MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2019/2020

TOP2121 – OBJECT-ORIENTED PROGRAMMING

(All sections / Groups)

12 OCTOBER 2019
2.30p.m – 4.30p.m
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This question paper consists of 8 pages, including the cover page, with **FOUR** questions only.

2. Attempt all **FOUR** questions.

3. All questions carry equal marks and the distribution of the marks for each question is given.

4. Please write all your answers in the answer booklet provided.

Question 1

(a) A class is an "abstraction" for a bundle of data and operations. Define what is abstraction and how can it be implemented with Java?

[2 marks]

(b) C is a procedural language whereas Java is an object-oriented language. State **TWO (2)** differences between the procedural language and object-oriented language. You can describe your answer with the aid of a diagram.

[4 marks]

(c) Java can be run on many different platforms. Describe with a diagram how the portability of Java code is achieved.

[4 marks]

Continued...

Question 2

(a) Identify **FOUR (4)** syntax errors in the Java code as shown in Figure 1.

```
import java.awt.*;
import java.util.*;
public class TOP2121{
   public void paint(g){
      Font f=new Font("Arial",Font.BOLD,20);
      g.setFont(f);
      g.setColor(C.blue);
      g.drawString("Welcome to the world of Java Applets",20,40);
      g.drawString("Today is "+ new Date(),20,120);
      }
   }
```

Figure 1. Erroneous Java Code

[4 marks]

(b) Based on given the code in Figure 2, answer the following questions:

i. Identify the default constructor from the code given in Figure 2.
ii. Discuss **THREE (3)** differences between the constructors and methods.

```
// Circle class, saved as "Circle.java"
public class Circle {
   // private variables
   private double radius;
   private String color;
   // constructors (overloaded)
   public Circle() {
      radius = 1.0;
      color = "red";
   }
   public Circle(double r) {
      radius = r;
      color = "red";
   }
   public Circle(double r, String c) {
      radius = r;
      color = c;
   }
```

Figure 2. A Sample Java Code

[4 marks]

Continued...

(c) In the Java programming language, *static* keyword can be used with block, variable, method and nested classes. In the following Java code as shown in Figure 3, there are **TWO (2)** errors. Identify the errors and use the *static* keyword to solve the errors.

```
class Student{
     String name;
     static String college = "ITS";

     static void change(){
          college = "KDU";
          name = "Fatimah";
          display();
     }
         void display(){
         System.out.println(name+" "+college);
   }
}
```

Figure 3. Erroneous Java Code

[2 marks]

Continued...

Question 3

(a) A Use Case Diagram is a graphic depiction of the interactions among the elements of a system. Design a Use Case diagram, with the following ATM system descriptions:

- There are 3 different users: Customer, Technician, and Bank.
- Customer is allowed to check balances, deposit funds, withdraw cash, and transfer funds.
- Technician is responsible to maintain and repair the ATM system.
- Bank is granted the access to monitor all activities performed by customer and technician.

[4 marks]

(b) The following figures illustrated a pair of classes for each case.

i) Shape Rectangle

ii) Car Steering Wheel

Determine the most appropriate relationship between the pair of classes and draw a connection between them using the corresponding UML symbol for each case.

[2 marks]

(c) Draw a class diagram that reflect the following Java code segment as shown in Figure 4.

```
public abstract class Vehicle {
      private int numberOfWheels;
      public Vehicle () { this(0); }
      protected Vehicle (int w) { numberOfWheels = w; }
      public int getNumberOfWheels () { return numberOfWheels; }
      public abstract boolean hasEngine ();
}
public class Car extends Vehicle {
      ...
}
public class Bicycle extends Vehicle {
      ...
}

public class MotorCycle extends Bicycle {
      ...
}
```

Figure 4. A Sample Java Code

[4 marks]

Continued...

Question 4

(a) Draw the output of the Java codes displayed in Figure 5.

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;

public class Exam extends Applet{
   Button b1=new Button("Final Exam");
   Button b2=new Button("Sup. Exam");
   Label l1=new Label("Exam Venue:");
   TextField tf1 = new TextField("Exam Hall");

   public void init(){
     add(l1);
     add(tf);
     add(b1);
     add(b2);
   }
}
```

Figure 5. A Java Applet Code.

[4 marks]

Continued...

(b) What is the output produced from the Java code as shown in Figure 6.

```
public class MMU
{
  public static void main(String[] args)
  {
     President a = new President();
     VicePresidentAcademic b = new VicePresidentAcademic();
     VicePresidentResearch c= new VicePresidentResearch();
     b.display();
     c.display();
  }
}
class President
{
  public void display()
  {
      System.out.println("President");
  }
  public void display(String s)
  {
      System.out.println(s);
  }
}
class VicePresidentAcademic extends President
{
  public void display()
  {
    super.display();
    System.out.println("Vice President Academic");
  }
}
class VicePresidentResearch extends President
{
  public void display()
  {
    super.display();
    System.out.println("Vice President Research");
  }
}
```

Figure 6. A Sample Java Code

[2 marks]

Continued...

(c) *FileContentDisplay* is a Java class that takes file path as the program parameter and display the content. You are required to complete the Java codes indicated by (i), (ii), (iii) and (iv). The example output is as depicted in Figure 7.

```
C:\Users\TOP2121>javac FileContenDisplay.java

C:\Users\TOP2121>java FileContentDisplay C:\Users\TOP2121\TextFile.txt
TOP2121, Object Oriented Programming sample file to read content.

C:\Users\TOP2121>
```

Figure 7. Example Output of `FileContentDisplay`

[4 marks]

```
import java.io.*;

public class FileContentDisplay {
   public static void main(String args[]) (i) _____ {
      int numRead = 0;
      FileReader reader = null;
      PrintWriter writer = null;
      char buffer[] = new char[512];

      if (args.length < 1) {
            System.out.println("Usage: java filename");
            System.exit(0);
      }
      try {
            reader = new FileReader (args [0]);
            writer = new PrintWriter(System.out);

            while ((numRead = (ii) _____ != -1){
            writer.write(buffer, 0, numRead);
            }
      }
      catch ((iii) _____ fe){
            System.out.println(fe.getMessage());
            System.exit(0);
      }
      catch ((iv) _____ ioe){
            System.out.println("Error reading/writing file");
      }
      finally {
            reader.close();
            writer.close();
      }
   }
}
```

End of Paper